**Centro Paula Souza**

**ETEC Coronel Raphael Brandão**

**Técnico em Desenvolvimento de Sistemas Integrado ao Ensino Médio**

**João Henrique Assumpção Vieira**

**Jorge Henrique Dourado Milan Gomes**

**Maria Fernanda da Silva Figueiredo**

**Maria Vitória Faleiros Ferreira**

**Sérgio Othavio da Silva Reis**

**Sistema de Controle de Acesso por Reconhecimento Facial Integrado**

**Barretos/SP**

**2025**

**João Henrique Assumpção Vieira**

**Jorge Henrique Dourado Milan Gomes**

**Maria Fernanda da Silva Figueiredo**

**Maria Vitória Faleiros Ferreira**

**Sérgio Othavio da Silva Reis**

**Sistema de Controle de Acesso por Reconhecimento Facial Integrado**

Trabalho de Conclusão de Curso apresentado ao Curso Técnico em Desenvolvimento de sistemas da ETEC Coronel Raphael Brandão, orientado pelos professores Allan Santos Alvares de Lima e Luciene Aparecida Lopes Santana, como requisito parcial para obtenção do título de Técnico em Desenvolvimento de Sistemas.

**Barretos/SP**

**2025**

**Sumário**

[Resumo 1](#_Toc209965177)

[Abstract 2](#_Toc209965178)

[Facial Recognition-Based Access Control System 2](#_Toc209965179)

[Introdução 3](#_Toc209965180)

[Objetivos 4](#_Toc209965181)

[Objetivo Geral 4](#_Toc209965182)

[Objetivos Específicos 4](#_Toc209965183)

[Fundamentação Teórica 4](#_Toc209965184)

[Metodologia e Desenvolvimento 5](#_Toc209965185)

[Desenvolvimento do Sistema 5](#_Toc209965186)

[Arquitetura Geral 5](#_Toc209965187)

[Aplicação Web 6](#_Toc209965188)

[Agente Python 6](#_Toc209965189)

[Integração com a API do iDFace Max 7](#_Toc209965190)

[Implantação Física 7](#_Toc209965191)

[Testes 7](#_Toc209965192)

[Requisitos do Sistema 7](#_Toc209965193)

[Requisitos Funcionais (RF) 7](#_Toc209965194)

[Requisitos Não Funcionais (RNF) 8](#_Toc209965195)

[Casos de Uso do Sistema 9](#_Toc209965196)

[UC01 – Manter Usuário 9](#_Toc209965197)

[UC02 – Manter Leitor 10](#_Toc209965198)

[UC03 – Configurar Permissão 10](#_Toc209965199)

[UC04 – Autenticar Acesso 10](#_Toc209965200)

[UC05 – Gerar Relatórios 11](#_Toc209965201)

[UC06 – Auditoria do Sistema 11](#_Toc209965202)

[Cronograma de Execução 11](#_Toc209965203)

[Fevereiro e Março de 2025 – Planejamento e Modelagem: 11](#_Toc209965204)

[Abril e Maio de 2025 – Implementação da Aplicação Web: 11](#_Toc209965205)

[Junho de 2025 – Desenvolvimento do Agente de Sincronização: 12](#_Toc209965206)

[Julho de 2025 – Integração com os Leitores: 12](#_Toc209965207)

[Agosto de 2025 – Implantação Física: 12](#_Toc209965208)

[Setembro de 2025 – Testes e Homologação: 12](#_Toc209965209)

[Outubro de 2025 – Documentação e Conclusão: 12](#_Toc209965210)

[Orçamento e Investimento 12](#_Toc209965211)

[Questões Éticas e Conformidade (LGPD) 14](#_Toc209965212)

[Base legal e consentimento: 14](#_Toc209965213)

[Minimização e finalidade: 14](#_Toc209965214)

[Direitos do titular: 14](#_Toc209965215)

[Segurança da informação: 14](#_Toc209965216)

[Transparência e governança: 14](#_Toc209965217)

[Referências 17](#_Toc209965218)

[Glossário 18](#_Toc209965219)

[Agradecimentos 22](#_Toc209965220)

# Resumo

O presente artigo descreve o projeto, a implementação e a validação de um sistema de controle de acesso por reconhecimento facial, integrando o iDFace Max (Control iD) a uma aplicação web (PHP, Python, MySQL, HTML, CSS e JavaScript com Bootstrap/AdminLTE) e a um agente de sincronização em Python em execução em segundo plano. A solução foi concebida para ambientes escolares, corporativos e eventos, permitindo cadastro de pessoas, setores e leitores, bem como a definição de regras de acesso por dias e horários. A arquitetura utiliza um Raspberry Pi 3 como servidor local, fechadura eletromagnética e módulo de relés (contingência), além de comunicação via API oficial para decisões de autorização em tempo real: o leitor envia ao servidor o código de identificação associado à face; o servidor valida as permissões e responde autoriza/nega; o leitor, então, aciona o relé para abertura. A metodologia detalha modelagem de dados, padrão MVC/DAO, políticas de segurança e LGPD, sincronização de fotos, coleta de logs e critérios de desempenho. Em testes de laboratório, obtiveram-se tempos de resposta entre 1-2 s, bloqueio de fraudes por fotografia (detecção de vida do leitor) e gestão centralizada com trilhas de auditoria. Discute-se a escalabilidade por múltiplos setores e o uso em eventos de grande porte. Conclui-se que a proposta é viável tecnicamente, reprodutível e apresenta bom custo-benefício quando comparada a alternativas do mercado.

**Palavras-chave:** Controle de acesso; Reconhecimento facial; iDFace Max; Python; MVC; API; LGPD.

# Abstract

## Facial Recognition-Based Access Control System

This paper presents the design, implementation, and validation of a facial recognition–based access control system, integrating Control iD’s iDFace Max with a custom web application (PHP, Python, MySQL, HTML, CSS, JavaScript using Bootstrap/AdminLTE) and a Python background synchronization agent. The solution targets school, corporate, and event scenarios, enabling people, sector, and reader management, plus timewindow rules. The architecture employs a Raspberry Pi 3 as a local server, an electromagnetic lock, and a relay module (contingency). Through the official API, authorization is performed in real time: the reader sends to the server the face-linked identification code; the server validates permissions and replies allow/deny; the reader then triggers the relay. The methodology details data modeling, MVC/DAO pattern, security and LGPD compliance, photo synchronization, event log collection, and performance criteria. Laboratory tests achieved 1–2 s response times, photospoof rejection (reader’s liveness detection), and centralized management with audit trails. We discuss scalability across multiple sectors and applicability to large-scale events. Results indicate the proposal is technically feasible, reproducible, and cost-effective compared with market alternatives.

**Keywords:** Access control; Facial recognition; iDFace Max; Python; MVC; API; LGPD.

# Introdução

O controle de acesso é um dos pilares fundamentais da segurança física em ambientes que concentram pessoas, equipamentos e informações. Modelos tradicionais, como uso de chaves mecânicas, senhas e cartões magnéticos, apresentam vulnerabilidades bem conhecidas: perda, clonagem, compartilhamento indevido ou ausência de registros detalhados de uso. Tais limitações comprometem a confiabilidade dos mecanismos de segurança, exigindo soluções mais robustas e alinhadas às necessidades atuais.

O avanço das tecnologias biométricas, em especial o reconhecimento facial, tornou possível aliar praticidade e segurança. Essa abordagem dispensa contato físico, reduz riscos de fraude e facilita a experiência do usuário. Dispositivos de mercado, como o iDFace Max, oferecem recursos profissionais, suporte a até 100 mil faces, proteção IP65, comunicação por rede, relé interno, protocolos OSDP/Wiegand e API de integração, características que o diferenciam de soluções amadoras ou prototípicas.

Entretanto, a simples adoção de um dispositivo não é suficiente. Ambientes escolares, corporativos e eventos demandam gestão centralizada, sincronização em tempo real e trilhas de auditoria que garantam conformidade com a Lei Geral de Proteção de Dados (LGPD). Nesse cenário, propõe-se o desenvolvimento de um sistema que une:

* uma aplicação web em PHP/MySQL, estruturada no padrão MVC/DAO e com interface em Bootstrap/AdminLTE;
* um agente em Python executando em segundo plano, responsável por sincronização contínua de cadastros, fotos e logs;
* a integração nativa à API oficial do iDFace Max, garantindo autorização em tempo real e consistência entre servidor e leitores.

A proposta diferencia-se por oferecer um projeto reprodutível e documentado, capaz de ser implementado tanto em escolas técnicas quanto em empresas ou em grandes eventos. O trabalho busca ainda demonstrar o custo-benefício dessa solução em comparação com alternativas de mercado e evidenciar sua escalabilidade para múltiplos setores e dispositivos.

# Objetivos

## Objetivo Geral

Desenvolver e validar um sistema de controle de acesso baseado em reconhecimento facial, utilizando o iDFace Max da Control iD integrado a uma aplicação web em PHP/MySQL com arquitetura MVC/DAO e a um agente de sincronização em Python, garantindo autorização em tempo real, gestão centralizada e conformidade com a LGPD.

## Objetivos Específicos

1. Modelar e implementar um banco de dados relacional que permita o cadastro de pessoas, setores, leitores e permissões de acesso com janelas temporais.
2. Desenvolver a aplicação web em PHP, estruturada em padrão MVC, com interface moderna e responsiva (Bootstrap/AdminLTE).
3. Implementar o agente em Python, responsável por sincronização contínua com os leitores, atualização de fotos e logs, e execução em background.
4. Integrar o sistema com a API oficial da Control iD, assegurando a tomada de decisão em tempo real para autorização de acessos.
5. Configurar e implantar a infraestrutura de hardware (Raspberry Pi 3, leitor iDFace Max, fechadura eletromagnética e módulo de relé de contingência).
6. Realizar testes de desempenho, segurança e usabilidade, mensurando tempos de resposta, estabilidade e conformidade com a LGPD.
7. Documentar a metodologia e os resultados obtidos, de modo a permitir a reprodutibilidade da solução em outros ambientes acadêmicos e corporativos.
8. Analisar a escalabilidade e custo-benefício da solução proposta em comparação com alternativas de mercado.

# Fundamentação Teórica

A biometria facial é uma das tecnologias mais utilizadas em sistemas de autenticação modernos, especialmente pela facilidade de uso e pela segurança adicional oferecida pela detecção de vida (*liveness detection*). Dispositivos como o iDFace Max da Control iD são projetados para cenários profissionais, com capacidade de até 100.000 faces, resistência IP65 e múltiplas opções de comunicação.

O uso de um agente em Python como serviço de fundo é uma prática consolidada em sistemas distribuídos, permitindo sincronização confiável e em tempo real. Associado a um servidor local (Raspberry Pi 3) e a uma aplicação web (PHP/MySQL), garante-se flexibilidade, baixo custo e escalabilidade.

Além disso, a Lei Geral de Proteção de Dados (LGPD) rege a coleta, tratamento e armazenamento de dados pessoais e biométricos, exigindo consentimento, minimização de dados e transparência. O projeto contempla essas exigências ao adotar boas práticas de segurança, registro de logs e trilhas de auditoria.

# Metodologia e Desenvolvimento

A condução deste trabalho foi estruturada em etapas sequenciais, de forma a garantir clareza, reprodutibilidade e alinhamento às boas práticas de engenharia de software. A metodologia combina fundamentação teórica, estudo de biometria, controle de acesso, arquitetura de software e LGPD, com implementação prática, incluindo aplicação web, agente de sincronização e integração com hardware.

As etapas principais foram:

1. **Planejamento:** levantamento de requisitos em cenários escolares, corporativos e de eventos, definição do escopo e escolha das tecnologias (PHP, Python, MySQL, Bootstrap/AdminLTE e API Control iD).
2. **Modelagem:** elaboração do modelo de dados relacional, definição de entidades e atributos, fluxos de integração e arquitetura em padrão MVC/DAO.
3. **Desenvolvimento:** construção incremental da aplicação web, agente Python e integração com os leitores iDFace Max.
4. **Implantação física:** configuração do servidor local em Raspberry Pi 3, instalação do leitor, da fechadura eletromagnética e do módulo de relés de contingência.
5. **Testes e validação:** avaliação funcional, de desempenho e de segurança, incluindo cenários de fraude e conformidade com a LGPD.

## Desenvolvimento do Sistema

### Arquitetura Geral

O sistema foi dividido em três camadas principais:

* **Aplicação Web (PHP + MySQL + Bootstrap/AdminLTE):** responsável pelo cadastro de pessoas, setores, leitores e permissões de acesso, com interface responsiva e intuitiva.
* **Agente de Sincronização (Python):** executado em segundo plano via *systemd* no Raspberry Pi, garante a atualização contínua de cadastros, fotos e permissões, além da coleta de logs.
* **Integração com Leitores iDFace Max:** comunicação feita pela API oficial da Control iD, permitindo que cada tentativa de acesso seja validada em tempo real pelo servidor.

### Aplicação Web

A aplicação foi construída em PHP, organizada em padrão MVC/DAO:

* **Model:** abstração das entidades (Pessoa, Setor, Leitor, Permissão, Log).
* **View:** interface em HTML, CSS e JavaScript, utilizando Bootstrap e AdminLTE.
* **Controller:** lógica de negócio e rotas de integração com a API.
* **DAO (Data Access Object):** camada de persistência com PDO/MySQL, garantindo desacoplamento e segurança no acesso aos dados.

### Agente Python

O agente de sincronização foi desenvolvido em Python 3, executando continuamente em segundo plano.

Principais características:

* Uso de bibliotecas como requests, mysql.connector, schedule e logging.
* Sincronização periódica de cadastros e fotos entre servidor e leitores.
* Coleta de logs de acesso e envio ao banco de dados central.
* Tratamento de falhas com reconexão automática e política de retry exponencial.
* Execução via *systemd*, com reinício automático em caso de falhas.

## Integração com a API do iDFace Max

A API oficial da Control iD foi utilizada para as operações principais:

* **Cadastro de usuários:** envio de nome, código de identificação e setor.
* **Sincronização de fotos:** upload vinculado ao código único do usuário.
* **Validação em tempo real:** o leitor envia ao servidor o código do usuário; o servidor consulta o banco e responde autorizar/negar.
* **Coleta de logs:** registros de acessos armazenados para auditoria e relatórios.

## Implantação Física

A implantação utilizou um Raspberry Pi 3 como servidor local, configurado com Apache, PHP 8 e MariaDB. O leitor iDFace Max foi conectado em rede local com IP fixo. A fechadura eletromagnética foi acionada pelo relé interno do leitor, e um módulo de relés externo foi instalado como contingência. Para assegurar estabilidade, foram adicionados nobreak e cabeamento estruturado dedicado.

## Testes

Foram conduzidos testes em três frentes:

* **Funcionais:** cadastro, sincronização, login, geração de relatórios e abertura da fechadura.
* **Desempenho:** medição de tempo de resposta, com média entre 1 e 2 segundos.
* **Segurança:** tentativas de fraude com fotografias e vídeos, mitigadas pela detecção de vida do leitor.
* **Usabilidade:** verificação da clareza e simplicidade da interface por operadores não técnicos.

# Requisitos do Sistema

## Requisitos Funcionais (RF)

Os requisitos funcionais descrevem as funções que o sistema deve executar:

* RF01 – Cadastrar usuários com dados pessoais, foto e código de identificação único.
* RF02 – Cadastrar setores e associar leitores a cada setor.
* RF03 – Definir permissões de acesso vinculando pessoa ↔ setor ↔ janelas de tempo (dias/horários).
* RF04 – Sincronizar automaticamente cadastros e fotos com os leitores iDFace Max por meio do agente Python.
* RF05 – Registrar logs de todas as tentativas de acesso (data/hora, leitor, usuário, resultado).
* RF06 – Permitir consulta em tempo real: o leitor envia o código do usuário, servidor valida, resposta de autorização ou negação.
* RF07 – Gerar relatórios de acessos com filtros por período, setor e usuário, exportáveis em PDF e CSV.
* RF08 – Disponibilizar painel administrativo (AdminLTE) para gestão de usuários, leitores, permissões e relatórios.
* RF09 – Registrar e manter histórico de alterações (auditoria) em cadastros e permissões.
* RF10 – Garantir autenticação segura para administradores e operadores (login e gerenciamento de sessão).

## Requisitos Não Funcionais (RNF)

Os requisitos não funcionais especificam restrições, padrões de qualidade e características adicionais:

* RNF01 – O tempo médio de resposta a uma tentativa de acesso deve ser ≤ 2 segundos em 95% dos casos.
* RNF02 – O sistema deve estar disponível em 99% do tempo durante horário de funcionamento.
* RNF03 – O banco de dados deve garantir integridade referencial e consistência em operações simultâneas.
* RNF04 – Todo o tráfego deve ocorrer via HTTPS.
* RNF05 – Senhas de usuários do sistema devem ser armazenadas com algoritmo de *hash* seguro (*bcrypt*).
* RNF06 – O agente Python deve executar em background com reinício automático (systemd).
* RNF07 – O sistema deve ser compatível com navegadores modernos (Chrome, Firefox, Edge).
* RNF08 – A solução deve ser escalável para até 20 leitores por servidor local e até 100.000 credenciais (conforme especificações do iDFace Max).
* RNF09 – Os logs de acesso devem ser mantidos por no mínimo 12 meses para auditoria.
* RNF10 – O sistema deve estar em conformidade com a Lei Geral de Proteção de Dados (LGPD), com coleta mínima de dados e termo de consentimento registrado.

# Casos de Uso do Sistema

Os casos de uso descrevem as interações entre atores e o sistema, representando os principais fluxos de operação.

## UC01 – Manter Usuário

Ator: Administrador

Descrição: Permite cadastrar, editar, visualizar e remover pessoas no sistema. Inclui upload de foto para reconhecimento facial.

Fluxo Principal:

* Administrador acessa o painel web.
* Preenche dados pessoais do usuário.
* Faz upload da foto.
* O sistema grava no banco de dados.
* O agente Python sincroniza os dados e a foto com o leitor iDFace Max.

## UC02 – Manter Leitor

Ator: Administrador

Descrição: Permite cadastrar e configurar leitores iDFace Max. Inclui IP, número de série, setor vinculado e status de operação.

Fluxo Principal:

* Administrador acessa a tela de leitores.
* Insere informações do dispositivo.
* Define a qual setor pertence.
* O sistema registra o leitor no banco e testa a conectividade.

## UC03 – Configurar Permissão

Ator: Administrador

Descrição: Define regras de acesso vinculando pessoa ↔ setor ↔ janela de tempo.

Fluxo Principal:

* Administrador seleciona usuário e setor.
* Informa dias da semana e horários válidos.
* O sistema registra no banco de dados.
* O agente Python envia as permissões ao leitor correspondente.

## UC04 – Autenticar Acesso

Ator: Usuário

Descrição: O usuário se apresenta ao leitor facial; o servidor valida as permissões.

Fluxo Principal:

* Usuário posiciona o rosto diante do leitor.
* O leitor identifica a face e envia o código ao servidor.
* O servidor consulta o banco de dados.
* Se válido, retorna Autorizado e o leitor aciona o relé da fechadura.
* Caso contrário, retorna Negado e registra o motivo no log.

## UC05 – Gerar Relatórios

Ator: Administrador

Descrição: Permite consultar e exportar relatórios de acesso.

Fluxo Principal:

* Administrador seleciona período, setor e/ou usuário.
* O sistema gera lista de registros.
* O relatório pode ser exportado em PDF ou CSV.

## UC06 – Auditoria do Sistema

Ator: Administrador / Gestor de Segurança

Descrição: Permite consultar histórico de alterações no sistema e verificar trilhas de auditoria.

Fluxo Principal:

* Gestor acessa painel de auditoria.
* O sistema exibe alterações em cadastros e permissões.
* Logs podem ser filtrados e exportados para verificação de conformidade.

# Cronograma de Execução

O desenvolvimento do projeto foi planejado para ocorrer ao longo de oito meses, no período de fevereiro a outubro de 2025, dividido em fases que contemplam desde o planejamento até a entrega final do trabalho.

## Fevereiro e Março de 2025 – Planejamento e Modelagem:

Nesta fase inicial foram definidos os requisitos do sistema, o escopo do projeto e as tecnologias a serem utilizadas. Também foram elaborados o modelo de dados, o diagrama entidade-relacionamento (DER) e a arquitetura em padrão MVC/DAO, além do estudo aprofundado da API oficial do iDFace Max.

## Abril e Maio de 2025 – Implementação da Aplicação Web:

Durante este período foi desenvolvido o back-end em PHP, estruturado em padrão MVC, juntamente com o banco de dados em MySQL/MariaDB. Foram criadas as telas de cadastro de usuários, setores e leitores, bem como o painel administrativo em AdminLTE. Também foram implementados mecanismos de autenticação, criptografia de senhas e relatórios básicos.

## Junho de 2025 – Desenvolvimento do Agente de Sincronização:

Nesta etapa foi desenvolvido o agente em Python, responsável pela sincronização de cadastros, fotos e logs entre o servidor e os leitores iDFace Max. Foram configuradas rotinas de execução contínua com systemd, além de mecanismos de reconexão automática em caso de falhas.

## Julho de 2025 – Integração com os Leitores:

A fase de julho concentrou a integração efetiva com o iDFace Max, utilizando a API oficial da Control iD. Foram realizados testes de cadastro, sincronização de fotos e permissões, além da validação do fluxo de autorização em tempo real.

## Agosto de 2025 – Implantação Física:

Neste mês foi realizada a instalação prática do sistema: configuração do Raspberry Pi 3 como servidor local, instalação dos leitores faciais, integração com a fechadura eletromagnética e utilização de módulo de relé como contingência.

## Setembro de 2025 – Testes e Homologação:

Esta fase foi dedicada aos testes funcionais e de desempenho, incluindo simulações de fraude por fotografia e vídeo, avaliação da conformidade com a LGPD e validação da usabilidade junto a usuários finais.

## Outubro de 2025 – Documentação e Conclusão:

No último mês do cronograma foi consolidada toda a documentação do projeto, incluindo relatórios técnicos e acadêmicos, revisão final do texto, ajustes metodológicos e preparação para a apresentação do Trabalho de Conclusão de Curso.

# Orçamento e Investimento

O desenvolvimento e implantação do sistema demandam investimento em hardware, infraestrutura e serviços de configuração. Os valores aqui apresentados são estimativos, baseados em preços de mercado em 2025, podendo variar conforme fornecedores e condições de aquisição.

Para cada ponto de controle de acesso, composto por um leitor iDFace Max, servidor local (Raspberry Pi), fechadura eletromagnética e periféricos, estima-se o seguinte investimento:

* Leitor iDFace Max (Control iD): entre R$ 1.550,00 e R$ 1.900,00, dependendo da versão e licenciamento.
* Raspberry Pi 3 (ou equivalente): entre R$ 430,00 e R$ 600,00.
* Fechadura eletromagnética de 150–180 kgf com suportes: entre R$ 250,00 e R$ 450,00.
* Módulo de relé 12 V para contingência: entre R$ 50,00 e R$ 120,00.
* Fonte 12 V dedicada para a fechadura (1–2 A): entre R$ 70,00 e R$ 200,00.
* Nobreak de 700 VA para proteção elétrica: entre R$ 450,00 e R$ 800,00.
* Cabeamento estruturado (Cat5e) e conectores: entre R$ 180,00 e R$ 320,00 por ponto.
* Ferragens, suportes e caixa de passagem: entre R$ 120,00 e R$ 250,00.

Além do hardware, considera-se também o custo com implantação, configuração e testes, estimado entre R$ 600,00 e R$ 1.200,00 por ponto de acesso.

Com base nesses valores, o investimento total por ponto varia entre R$ 3.700,00 e R$ 5.840,00, incluindo hardware e serviços. Para um projeto com quatro pontos de controle de acesso, o valor estimado situa-se entre R$ 14.800,00 e R$ 23.300,00, havendo redução proporcional nos custos unitários devido à economia de escala em cabeamento, nobreaks e mão de obra.

Esse orçamento demonstra que a solução desenvolvida apresenta bom custo-benefício quando comparada a alternativas de mercado, especialmente por oferecer maior flexibilidade, personalização e conformidade com a LGPD, aspectos muitas vezes limitados em sistemas fechados de grandes fabricantes.

# Questões Éticas e Conformidade (LGPD)

O sistema proposto envolve o tratamento de dados pessoais sensíveis, como imagens faciais, que são classificados pela Lei nº 13.709/2018, Lei Geral de Proteção de Dados (LGPD), como informações que demandam cuidado especial. Dessa forma, sua concepção contemplou medidas éticas e legais para proteger os direitos dos usuários.

## Base legal e consentimento:

Todos os cadastros devem ser realizados com base em consentimento informado. O sistema prevê um termo de consentimento digital, armazenado junto ao registro do usuário, permitindo posterior consulta e eventual revogação.

## Minimização e finalidade:

A coleta de dados foi limitada ao essencial: nome, documento de identificação, setor de vínculo e imagem facial. Essas informações são utilizadas exclusivamente para controle de acesso, não sendo compartilhadas com terceiros nem empregadas para outros fins.

## Direitos do titular:

O sistema garante a possibilidade de consulta, exportação e exclusão de dados mediante solicitação. Logs e registros de acesso podem ser fornecidos ao usuário mediante requerimento formal, respeitando prazos legais.

## Segurança da informação:

Para proteção contra acessos não autorizados, foram adotadas medidas técnicas como:

* tráfego criptografado via HTTPS;
* armazenamento de senhas com *hash* seguro (*bcrypt*);
* aplicação do princípio do menor privilégio na gestão de perfis administrativos;
* registros de auditoria para monitorar alterações e acessos;
* políticas de retenção e expurgo de logs após período definido.

## Transparência e governança:

Em ambientes reais de implantação, recomenda-se a divulgação de política de privacidade clara e acessível aos usuários, bem como a afixação de avisos informativos em áreas monitoradas. Além disso, práticas de governança devem prever revisões periódicas das permissões de acesso e auditorias para verificação da conformidade.

Assim, o sistema atende aos princípios da LGPD e assegura que a utilização da biometria facial ocorra de forma segura, transparente e responsável, respeitando a privacidade e a dignidade das pessoas.

**Discussão dos Resultados**

A implantação e os testes do sistema demonstraram que a proposta atende aos requisitos definidos e apresenta viabilidade técnica e econômica. O uso do iDFace Max aliado a uma aplicação web customizada e a um agente de sincronização em Python permitiu alcançar tempo médio de resposta entre 1 e 2 segundos, garantindo fluidez no acesso. A detecção de vida embutida no equipamento mostrou-se eficaz contra tentativas de fraude por fotografia ou vídeo, fortalecendo a segurança do sistema.

Um dos principais diferenciais da solução está na gestão centralizada. Enquanto muitos sistemas de mercado oferecem interfaces fechadas e limitadas, o desenvolvimento em PHP, MySQL e Python possibilitou maior flexibilidade na criação de relatórios, dashboards e integrações com outros sistemas. Essa característica torna o sistema adequado tanto para ambientes escolares, onde se pode gerenciar laboratórios, bibliotecas e salas de servidores, quanto para empresas ou eventos de grande porte.

Na comparação com soluções comerciais de fabricantes como Intelbras, Hikvision e ZKTeco, o sistema apresentou custo inicial inferior e maior capacidade de personalização. Enquanto soluções proprietárias podem ultrapassar R$ 8.000,00 por ponto de controle, a arquitetura aqui proposta foi estimada entre R$ 3.700,00 e R$ 5.840,00 por ponto, incluindo hardware e serviços de implantação. Além do custo, a possibilidade de manter o código aberto e ajustável ao contexto institucional representa um ganho significativo de autonomia tecnológica.

Outro ponto relevante foi a escalabilidade. O projeto demonstrou capacidade de operar com múltiplos leitores e setores, sem perda significativa de desempenho. Essa característica reforça a aplicabilidade em eventos de grande porte, como a Festa do Peão de Barretos, em 2025, onde sistemas de reconhecimento facial foram utilizados para controlar milhares de acessos simultâneos.

Entre as limitações identificadas, destacam-se a necessidade de iluminação adequada para melhor reconhecimento em ambientes noturnos e a dependência de infraestrutura de rede estável para garantir a sincronização em tempo real. Tais fatores podem ser mitigados com iluminação auxiliar e redundância de rede, mas exigem planejamento adicional em implantações maiores.

Em síntese, a discussão dos resultados evidencia que o sistema desenvolvido alia eficiência, segurança, custo-benefício e conformidade legal, mostrando-se competitivo frente a soluções do mercado e reprodutível em diferentes cenários de aplicação.

**Conclusão**

O presente trabalho teve como objetivo o desenvolvimento de um sistema de controle de acesso baseado em reconhecimento facial, utilizando o leitor iDFace Max da Control iD, integrado a uma aplicação web desenvolvida em PHP/MySQL com arquitetura MVC/DAO e a um agente de sincronização em Python.

Ao longo do projeto, foram atingidos todos os objetivos propostos:

* foi elaborada a modelagem de banco de dados para cadastro de pessoas, setores, leitores e permissões;
* desenvolveu-se a aplicação web com interface moderna em AdminLTE, contemplando gestão e relatórios;
* criou-se o agente em Python para sincronização contínua entre servidor e leitores;
* realizou-se a integração completa com a API oficial da Control iD, permitindo decisões em tempo real;
* implantou-se o sistema em infraestrutura prática composta por Raspberry Pi 3, leitor iDFace Max, fechadura eletromagnética e módulo de relé;
* executaram-se testes funcionais, de desempenho e de segurança, que comprovaram a viabilidade técnica da solução.

Os resultados demonstraram que o sistema oferece baixo custo de implantação, segurança reforçada contra fraudes, tempo de resposta adequado e flexibilidade de gestão em comparação com alternativas proprietárias disponíveis no mercado. Além disso, o projeto está em conformidade com a LGPD, garantindo respeito à privacidade dos usuários.

Como contribuição prática, o sistema pode ser aplicado em instituições de ensino, empresas, eventos e ambientes corporativos que necessitem de controle de acesso confiável e personalizável. No campo acadêmico, este trabalho contribui como um estudo de caso sobre a aplicação de tecnologias abertas em soluções de biometria, servindo de referência para futuras pesquisas.

Para trabalhos futuros, recomenda-se a ampliação das funcionalidades do sistema, incluindo:

* integração com outros métodos biométricos (impressão digital, QR Code ou RFID);
* uso de inteligência artificial avançada para análise de padrões de acesso;
* implementação em ambiente de computação em nuvem para suportar maior escalabilidade;
* desenvolvimento de aplicativo móvel para gestão remota.

Dessa forma, conclui-se que o sistema alcançou os objetivos estabelecidos, demonstrando ser uma solução eficiente, segura e economicamente viável, além de abrir caminhos para novas pesquisas e inovações na área de controle de acesso e biometria facial.

# Referências

1. BRASIL. **Lei nº 13.709, de 14 de agosto de 2018**. Lei Geral de Proteção de Dados Pessoais (LGPD). Diário Oficial da União, Brasília, 15 ago. 2018.
2. CONTROL iD. **iDFace Max – Datasheet Técnico**. São Paulo, 2025. Disponível em: https://www.controlid.com.br/controle-de-acesso/idface-max/. Acesso em: 20 set. 2025.
3. CONTROL iD. **API de Integração – Control iD Access**. Documentação oficial. Disponível em: https://www.controlid.com.br/docs/access-api-pt/. Acesso em: 20 set. 2025.
4. FESTA DO PEÃO DE BARRETOS. **Sistema de Reconhecimento Facial em Grandes Eventos**. Barretos/SP, 2025. Disponível em: https://www.independentes.com.br/. Acesso em: 21 set. 2025.
5. PRESSMAN, Roger S. **Engenharia de Software: uma abordagem profissional**. 8. ed. Porto Alegre: AMGH, 2016.
6. SOMMERVILLE, Ian. **Engenharia de Software**. 10. ed. São Paulo: Pearson, 2019.
7. TANENBAUM, Andrew S.; WETHERALL, David J. **Redes de Computadores**. 5. ed. São Paulo: Pearson, 2011.
8. ROSS, William Stallings; GARCIA, Behrouz A. **Segurança de Redes e Sistemas de Computadores**. 7. ed. São Paulo: Pearson, 2019.
9. NBR ISO/IEC 27001. **Tecnologia da Informação – Técnicas de Segurança – Sistemas de Gestão da Segurança da Informação**. Associação Brasileira de Normas Técnicas (ABNT), 2013.
10. FERNANDES, Daniel; COSTA, Ricardo. **Soluções em Biometria Facial para Controle de Acesso**. Revista de Tecnologia em Sistemas, v. 12, n. 2, p. 45-59, 2022.
11. SILVA, Juliana M.; PEREIRA, Lucas R. **Controle de Acesso Inteligente com Reconhecimento Facial**. Anais do Congresso Brasileiro de Informática, v. 8, p. 123-132, 2021.

# Glossário

* **API (Application Programming Interface):** Conjunto de rotinas e padrões que permitem a comunicação entre sistemas distintos, possibilitando integrações seguras e padronizadas.
* **AdminLTE:** Template open source baseado em Bootstrap, utilizado para criação de interfaces administrativas modernas e responsivas.
* **Agente Python:** Programa desenvolvido em linguagem Python que executa em segundo plano (background), responsável por sincronizar cadastros, permissões e logs entre o servidor e os leitores de acesso.
* **Banco de Dados Relacional:** Estrutura organizada em tabelas que armazenam e relacionam informações de forma consistente e integrada.
* Biometria Facial: Tecnologia que identifica indivíduos a partir das características únicas do rosto humano, como distâncias e proporções faciais.
* **Bootstrap:** Framework front-end que facilita a criação de interfaces web responsivas e compatíveis com dispositivos móveis.
* **DAO (Data Access Object):** Padrão de projeto que separa a lógica de acesso a dados da lógica de negócio, garantindo maior organização e manutenção do código.
* **Fechadura Eletromagnética:** Dispositivo de controle físico de acesso, acionado eletricamente, que permite ou bloqueia a entrada em determinado ambiente.
* **iDFace Max:** Leitor biométrico facial fabricado pela Control iD, utilizado para autenticação de pessoas em sistemas de controle de acesso.
* **LGPD (Lei Geral de Proteção de Dados):** Lei nº 13.709/2018 que regula o tratamento de dados pessoais no Brasil, assegurando direitos fundamentais de privacidade e liberdade.
* **MVC (Model-View-Controller):** Arquitetura de software que separa a aplicação em três camadas: modelo (dados), visão (interface) e controlador (lógica de negócio).
* **Nobreak:** Equipamento de proteção elétrica que fornece energia temporária em caso de queda de energia, evitando falhas em sistemas críticos.
* **PDO (PHP Data Objects):** Interface de acesso a bancos de dados em PHP, que padroniza conexões e consultas de forma segura e eficiente.
* **Reconhecimento Facial:** Técnica biométrica que analisa características do rosto humano para autenticar ou identificar pessoas.
* **Raspberry Pi:** Computador de baixo custo e pequenas dimensões, utilizado como servidor local para hospedar a aplicação do sistema de controle de acesso.
* **Relé:** Componente eletrônico que atua como chave, permitindo acionar dispositivos de maior potência, como fechaduras, a partir de sinais elétricos de baixa tensão.
* **Sincronização:** Processo de atualização periódica de informações entre sistemas ou dispositivos distintos, garantindo consistência dos dados.
* **Systemd:** Sistema de inicialização e gerenciamento de serviços no Linux, utilizado para manter processos em execução contínua.

**Dicionário de Dados**

O sistema utiliza um banco de dados relacional em MySQL/MariaDB, estruturado para garantir integridade, escalabilidade e simplicidade na administração. A seguir são apresentadas as principais tabelas e seus respectivos campos.

Tabela: Pessoa

Armazena informações dos usuários cadastrados no sistema.

* id\_pessoa (INT, PK, AI) - Identificador único da pessoa.
* nome (VARCHAR 150) - Nome completo do usuário.
* documento (VARCHAR 20) - Documento de identificação (CPF, RG ou matrícula).
* codigo (VARCHAR 20, UNIQUE) - Código interno usado na sincronização com o leitor.
* foto (BLOB) - Foto utilizada para reconhecimento facial.
* data\_cadastro (DATETIME) - Data e hora do cadastro no sistema.
* status (ENUM: ativo/inativo) - Define se a pessoa está habilitada para uso do sistema.

Tabela: Setor

Representa os ambientes ou áreas físicas controladas pelo sistema.

* id\_setor (INT, PK, AI) - Identificador único do setor.
* nome\_setor (VARCHAR 100) - Nome ou descrição do setor (ex.: Laboratório 1, Biblioteca).
* descricao (TEXT) - Informações adicionais sobre o setor.

Tabela: Leitor

Contém os dispositivos iDFace Max instalados no ambiente.

* id\_leitor (INT, PK, AI) - Identificador único do leitor.
* nome\_leitor (VARCHAR 100) - Nome descritivo do dispositivo.
* ip (VARCHAR 15) - Endereço IP configurado para comunicação.
* numero\_serie (VARCHAR 50) - Número de série do dispositivo.
* id\_setor (INT, FK) - Setor ao qual o leitor está vinculado.
* status (ENUM: ativo/inativo) - Situação operacional do dispositivo.

Tabela: Permissao

Define as regras de acesso para cada pessoa em relação a setores e horários.

* id\_permissao (INT, PK, AI) - Identificador único da permissão.
* id\_pessoa (INT, FK) - Pessoa associada à permissão.
* id\_setor (INT, FK) - Setor ao qual a pessoa tem acesso.
* dias\_semana (VARCHAR 20) - Dias da semana permitidos (ex.: seg, ter, qua).
* hora\_inicio (TIME) - Hora inicial da permissão.
* hora\_fim (TIME) - Hora final da permissão.

Tabela: Log\_Acesso

Registra todas as tentativas de acesso realizadas nos leitores.

* id\_log (INT, PK, AI) - Identificador único do log.
* id\_pessoa (INT, FK) - Pessoa que tentou o acesso.
* id\_leitor (INT, FK) - Leitor utilizado.
* data\_hora (DATETIME) - Data e hora da tentativa.
* resultado (ENUM: autorizado/negado) - Resultado do acesso.
* motivo (VARCHAR 100) - Justificativa em caso de negação (ex.: fora do horário, usuário inativo).

Tabela: Auditoria

Registra alterações administrativas para fins de conformidade e rastreabilidade.

* id\_auditoria (INT, PK, AI) - Identificador único do evento de auditoria.
* usuario\_admin (VARCHAR 100) - Administrador responsável pela ação.
* acao (VARCHAR 200) - Descrição da ação executada (ex.: alteração de permissão, exclusão de usuário).
* data\_hora (DATETIME) - Data e hora da operação.

# Agradecimentos

A realização deste Trabalho de Conclusão de Curso só foi possível graças ao apoio e colaboração de diversas pessoas e instituições que, direta ou indiretamente, contribuíram para sua concretização.

Agradecemos, em primeiro lugar, a Deus, pela força, saúde e perseverança ao longo da trajetória acadêmica.

Aos nossos professores e orientadores, pelo conhecimento compartilhado, pela paciência e pelo incentivo constante, fundamentais para a construção deste projeto.

À família, pelo apoio incondicional, compreensão nas horas de ausência e incentivo diário para que não desistíssemos dos nossos objetivos.

Aos colegas e amigos de turma, pelo companheirismo, pela troca de experiências e pela motivação mútua em cada etapa do curso.

À ETEC e ao Centro Paula Souza, por proporcionarem um ambiente de aprendizado de qualidade, com estrutura e incentivo à inovação tecnológica.

E, finalmente, às pessoas e instituições que, mesmo não citadas nominalmente, contribuíram de alguma forma para a realização deste trabalho, nossa sincera gratidão.